

Egzamin (2023) - Wstęp do programowania

- Zestaw 101

Zadanie 1: 7 pkt. Zadanie 2: 8 pkt. Zadanie 3: 9 pkt. Zadanie 4: 8 pkt. Zadanie 5: 9 pkt. Zadanie 6: 9p pkt.

Punktacja: 46-50 pkt - bdb(5,0); 41-45 pkt - db+(4,5); 36-40 pkt - db(4,0); 31-35 pkt - dst+(3,5); 26-30 pkt - dst(3,0); 0-25 pkt - ndst (2,0).

- Każde zadanie powinno być w oddzielnym projekcie.
- Kod zakomentowany nie będzie sprawdzany.
- Kod musi spełniać zasady kompilacji.

Zad.1. W folderze Debug na pendrive znajduje się projekt z kodem w języku C. W pliku main.cpp w niektórych liniijkach są komentarze. Twoim zadaniem jest wpisanie wartości odpowiednich zmiennych po wykonaniu konkretnej linii kodu. Dopisanie nowych liniijek i instrukcji poza komentarzami będzie powodowało 0 pkt.

Zad.2. Napisz program, który pobiera od użytkownika trzy liczby całkowite a, b, c . Program ma wyświetlić informację, czy liczby są uporządkowane nierosnąco w kolejności ich wprowadzania lub nie.

Zad.3. Napisz funkcję, która jako argument otrzymuje dodatnią liczbę całkowitą n i zwraca liczbę równą sumie liczb całkowitych dodatnich podzielnych przez 3 oraz mniejszych n . Nie korzystaj z żadnych gotowych funkcji bibliotecznych ani wbudowanych poza instrukcjami wejścia/wyjścia. Stwórz dwa przypadki testowe dla funkcji.

Zad.4. Napisz funkcję, która otrzymuje trzy argumenty: dodatnią liczbę całkowitą n , dodatnią liczbę całkowitą a oraz n -elementową tablicę `tab` o elementach typu `int`. Funkcja ma wyświetlić ile elementów tablicy `tab` jest dodatnią wielokrotnością a . W przypadku braku takich elementów wyświetl 0. Stwórz przypadek testowy dla funkcji.

Zad.5. Napisz program, który pobierze od użytkownika 5 liczb całkowitych, zapisze je do odpowiedniego wektora, a następnie wypisze ile elementów z nich jest ujemnych.

Zad.6. Napisz rekurencyjną funkcję zwracającą dla otrzymanej w argumencie nieujemnej liczby całkowitej n wartość elementu (liczbę całkowitą) o indeksie n ciągu zdefiniowanego w następujący sposób:

$$a_0 = a_1 = 1$$
$$a_n = a_0 + a_1 + \dots + a_{n-1} \text{ dla } n > 1$$